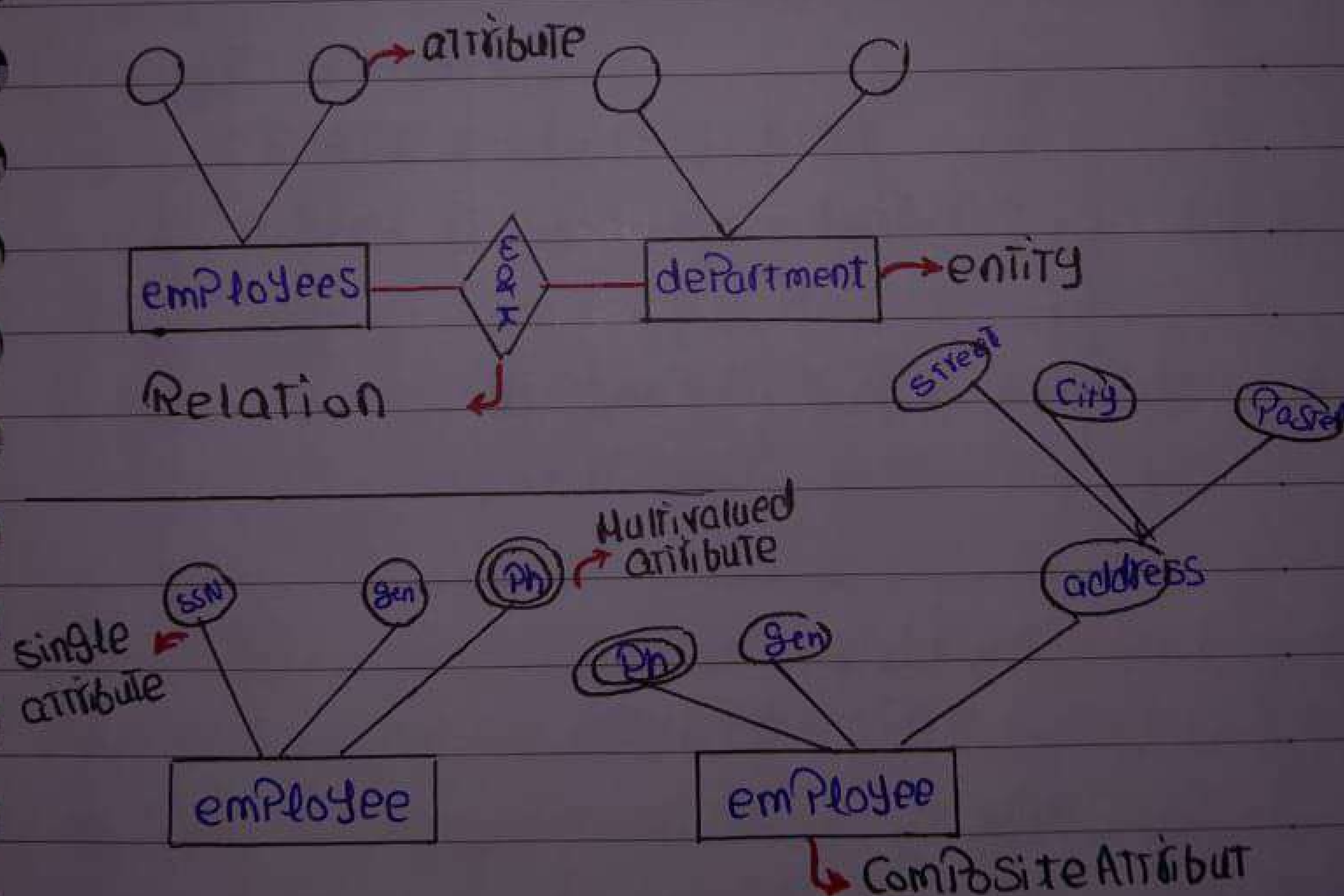
- sql ⇒ structure query language
- DDL ⇒ Data Definition language. ⇒ create Table
- DML ⇒ Data Manibulation language. ⇒ insert / delete
- Dcl ⇒ Data Control language ⇒ بيدي اليوزر صلاحيات معينة

⇒ attributes ⇒ بتكون ال Header

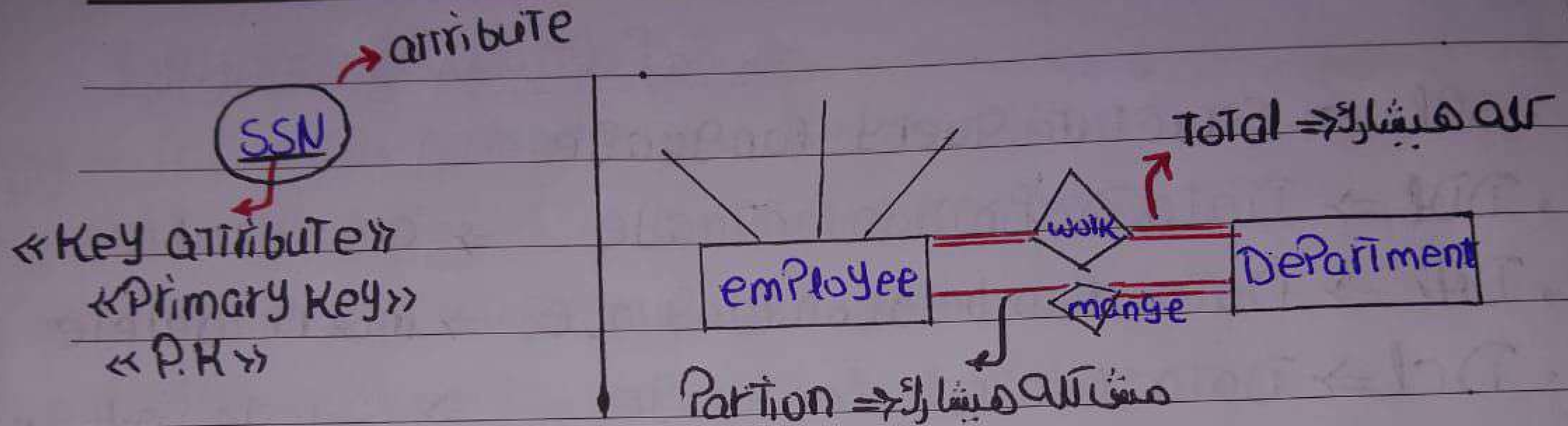
⇒ scema ⇒ شكل الجدول عامل ازاى

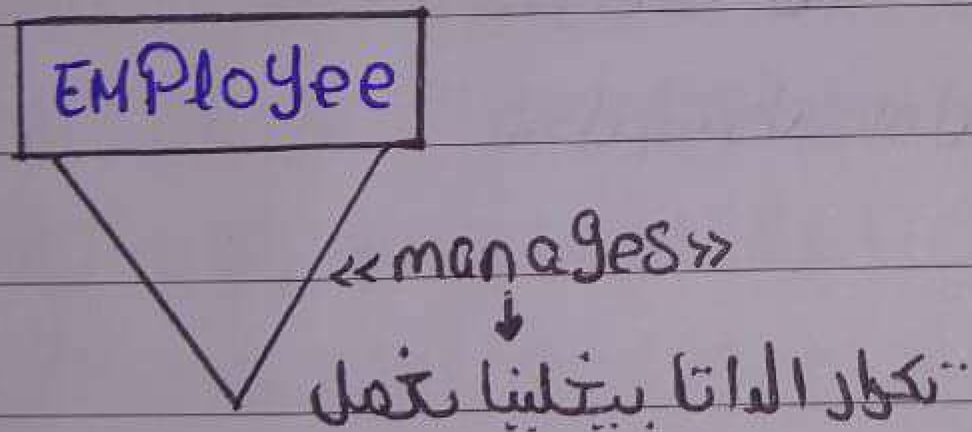
ARD ⇒ مودل



- Composite + Multivalued = " Complex "

الموضوع: التاريخ: / /





<<Recursion>> <<Self Join>>

⇒ using DDL statements to create data structure

1) Naming Rules

1) Must being char «letter» «لازم يبدأ بحرف»
2) 1-30 char
3) A-Z, a-z, 0-9, _
4) not duplicate ممنوع التكرار
5) not be reserved word. مش بنسمي أسماء محجوزة

DDL ⇒ data definition language

Create ⇒ Table

CREATE table [اسم التابل]
table(column dataType [ ]);

var char 2(), بنستخدم string ال ⇐
↓ نهاية للعمود

Constrains

- Not Null
- unique
- Primary key
- Foreign key
- Check

• column level constrain «الكونستريت على مستوى التابل»

```
CREATE TABLE employees(
employee_id NUMBER(6)
CONSTRAINT em_id_PK PRIMARY KEY,
first_name VARCAR2 (10),
----);
```

مختصر مباشرة

الموضوع: ________ التاريخ: / /

• table-level constraint

```
CREATE TABLE employees (
employee-id NUMBER(6),
first-name VARCHAR2(20),
...
job-id VARCHAR2(10) NOT Null,
CONSTRAINT emp-emp-id-PK
PRIMARY key ((EMPLOYEE-ID));
```

employ id ← لازم نزودها طالما مكتبناش الكونسترين تحت.

unique & check

Constraint F1 أي unique(email),
Constraint F1 si check(salary >70),

* constraint - fore-key(_)
REFERANCE
department (_);
m ← to 1

⇒ Alter , Modify , ADD , Rename , drop

مثلا (F1 الشكل) drop كده هيمسحه كله

Alter ↓ للتعديل
Modify ↓ تعديل الداتا تايب
ADD ↓ إضافة كونسترين
Rename ↓ من _ إلى _
drop ↓ مسح

← عشان البيت الـ (Pharmacy)
.جوا الأدمن واعمل يوزر جديد
Create user F1 اي
IDentified by F1 نفس الـ

insert الداتا ← إضافة الداتا في
الجدول قبل الـ insert بنحط
في النهاية بنحط BEGIN
END;



«Special Cases»

التاريخ: / /   الموضوع:

1)

Recursive Relation

↱ foriegn key

كـ الطبيعي انك بتعمل Constrain من جدول ال many

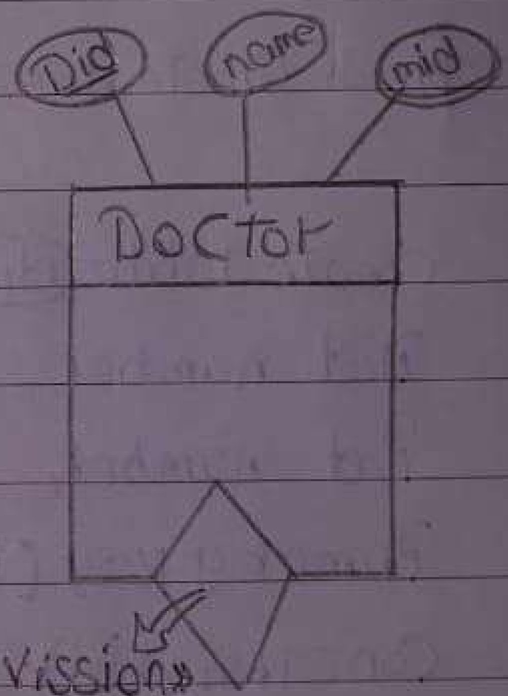
بيشاور على جدول ال one

كـ Recursive ⇐ هيشاور على نفسه

```
Create table Doctor (
Did integar Primary key,
Name VarChar2 (200),
Mid integar,
Constrain fk Mid foriegnkey (Mid) reference Doctor
 (Did)
);
```

Mid ↙ Manger

Doctor (Did)



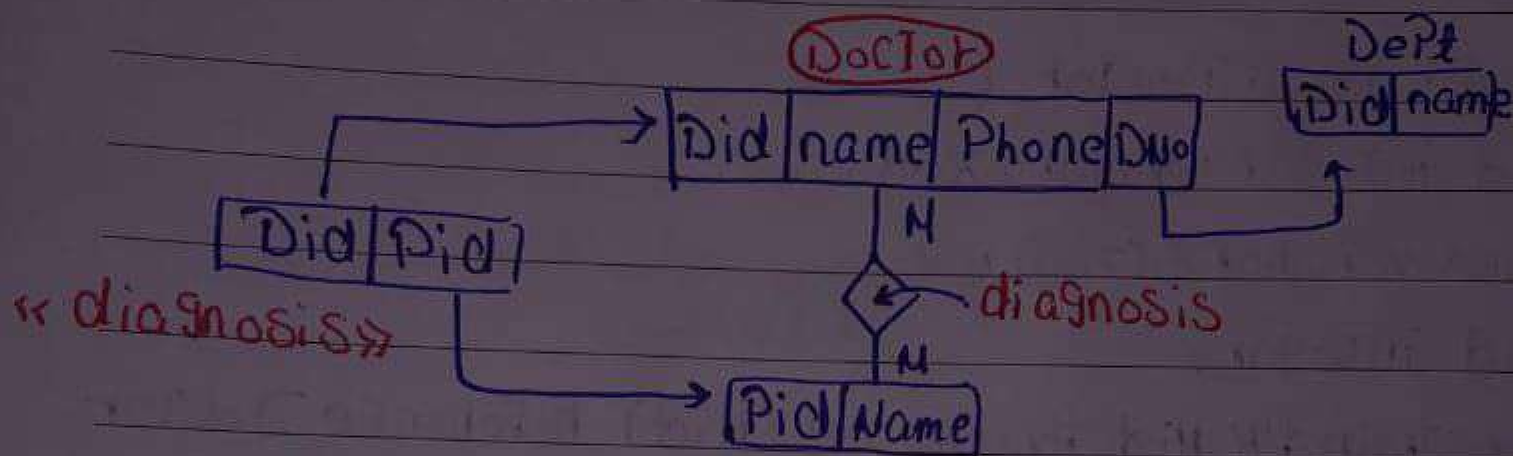
التاريخ: / /

الموضوع:

2)

«Many to Many»

مشاعندنا علاقة M To M عشان كدا بنعمل create عادي للـ 2 tables اللي عندك وبعدين نعمل جدول بيكون عبارة عن الـ attributes اللي هي P.K في الجدولين



• بعد ما هنعمل كود الـ 2 tables هنعمل create للـ table الجديد

```
Create Table diagnosis(
Did number,
Pid number,
Primary key (Did,Pid),
Constraint Dfk foriegn key (Did)
referance Doctor(Did),
Constraint PFK foriegn key (Pid)
referance Patient (Pid)
);
```

• لازم يكونوا نفس الداتا تايب لكن الاسم عادي «As You like»

التاريخ: / /

الموضوع:

3) Multi valued attributes

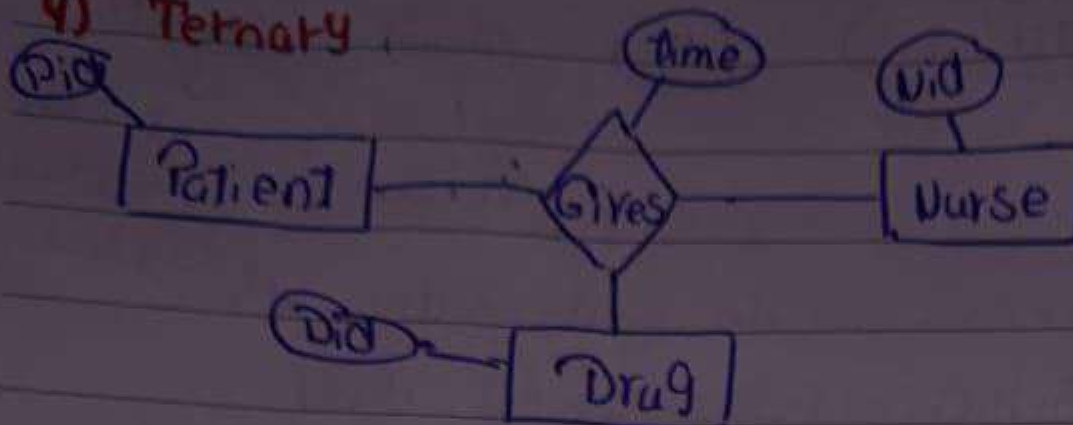


له أكتر من رقم تليفون

كدة بنعمل جدول منفصل بس لازم جدول ال Person يتعمل الأول بس مش هيكون فيه ال Phone

```
Create table Person(
Pid number Primary key,
name varchar2(200)
);
Create table Person_Phone(
Pid number;          // هنا ال Primary key
Phone varchar2(15);  بنعمله var مش num عشان يسجل ال Zero
Primary key (Pid, Phone)
Constraint Pid_FK forigen key (Pid) referances
Person (Pid)
);
```

4) Ternary



ك العلاقة ثلاثية لو مثلا عاوز أعرف الوقت اللى ال Nurse
هتدى فيه ال Patient ال drug فال time بيكون على
العلاقة نفسها فهنعمل 3 tables عادى لل entities
ولكن العلاقة الثلاثية بتحول لجدول.

```
Create Table Gives (
    Did Number,
    Nid Number,
    Did Number,
    time date,
    Primary Key (Pid, Did, Nid),
    Constraint Pid-FK Forigien Key (Pid) referance
    Patient (Pid),
        "  Nid-FK    "   "  (Nid)  "
    Nurse (Nid),
        "  Did-FK    "   "  (Did)  "
    Doctor (Did)
);
```

P.K بتوع الجداول (Did, Nid, Did)

time ← بتاع ال Gives

وكل واحد بشاور على جدول